

Title: Method for partitioning memory block and identifying R/W information in flash memory			
Application Number:	00129828	Application Date:	2000.10.27
Publication Number:	1351350	Publication Date:	2002.05.29
Approval Pub. Date:		Granted Pub. Date:	
International Classification:	G11C7/00,G11C16/06		
Applicant(s) Name:	Yue Jingxing		
Address:	100101		
Inventor(s) Name:	Yue Jingxing		
Attorney & Agent:	liu fang		
Abstract			
<p>A method for partitioning the memory block and identifying the R/W information in flash memory is disclosed. The memory blocks in the flash memory are divided into data region an control region. Said control region is further divided into index area for indexing if the byte information is used or not, logic address pointer area for pointing the physical address on logic adress of memory block, and state ID area. Its advantages include high data R/W speed and long service life.</p>			



[19] 中华人民共和国国家知识产权局

[51] Int. Cl<sup>7</sup>

G11C 16/06

G11C 7/00

## [12] 发明专利申请公开说明书

[21] 申请号 00129828.3

BEST AVAILABLE COPY

[43] 公开日 2002 年 5 月 29 日

[11] 公开号 CN 1351350A

[22] 申请日 2000.10.27 [21] 申请号 00129828.3

[71] 申请人 岳京星

地址 100101 北京市朝阳区北四环中路 33 号

共同申请人 吴秀林

[72] 发明人 岳京星 吴秀林

[74] 专利代理机构 北京三友知识产权代理有限公司  
代理人 刘芳

权利要求书 1 页 说明书 4 页 附图页数 2 页

[54] 发明名称 闪存中存储块的分区及读写信息标识的方法

[57] 摘要

一种闪存中存储块的分区及读写信息标识的方法，闪存内部各存储块划分为数据存储区和控制区，控制区划分为索引区、逻辑地址指针区和状态标识区。索引区索引位组信息为未用过或已使用；逻辑地址指针组信息为存储块的物理地址或按照更新次数顺序记录的逻辑地址。本发明通过控制区的合理划分和标识记录，有效地实现了对闪存片数据块读写信息的管理，从而能够将对数据块写入和擦除动作分别完成，提高了数据读写的速度，而且能够均匀地使用闪存片个数据块，延长其使用寿命。

INDEX	LOG	PHYS	STATUS	RESERVED
0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24
25	26	27	28	29
30	31	32	33	34
35	36	37	38	39
40	41	42	43	44
45	46	47	48	49
50	51	52	53	54
55	56	57	58	59
60	61	62	63	64
65	66	67	68	69
70	71	72	73	74
75	76	77	78	79
80	81	82	83	84
85	86	87	88	89
90	91	92	93	94
95	96	97	98	99

ISSN 1008-4274

知识产权出版社出版



001027

## 权 利 要 求 书

1、一种闪存中存储块的分区及读写信息标识的方法，闪存内部的各存储块（Block）均划分有数据存储区和控制区，其特征在于：该控制区划分为索引区、逻辑地址指针区和状态标识区。

2、如权利要求 1 所述的闪存中存储块的分区及读写信息标识的方法，其特征在于：所述的状态标识区标识所对应存储块工作状态的状态标识位组可为相对应的存储块（Block）的三种状态之一：“空”或“忙”或“释放”；“空”表示给数据块内没有数据，可以使用；“忙”表示该数据块已经被占用，不能再写入数据；“释放”表示该数据块可以被擦除。

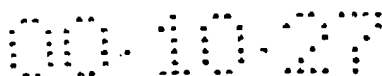
3、如权利要求 1 所述的闪存中存储块的分区及读写信息标识的方法，其特征在于：所述的索引区中的索引位组的信息为与之相对应的存储块使用状态之一：“未用过”或“已使用”；“未用过”表示数据块内未写入数据，“已使用”表示数据块内写入数据后。

4、如权利要求 1 所述的闪存中存储块的分区及读写信息标识的方法，其特征在于：所述的逻辑地址指针区内的逻辑地址指针组信息为与之相对应的存储块的物理地址或表示多次更新后、按照更新次数顺序排列记录的逻辑地址。

5、如权利要求 1 或 3 或 4 所述的闪存中存储块的分区及读写信息标识的方法，其特征在于：所述的索引区中的索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量相等。

6、如权利要求 1 或 3 或 4 所述的闪存中存储块的分区及读写信息标识的方法，其特征在于：所述的状态标识区的状态标识位组的数量可与数据块的数量相等。





## 说明书

### 闪存中存储块的分区及读写信息标识的方法

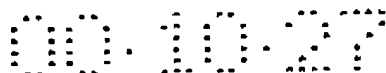
5 本发明涉及一种闪存 (Flash memory) 芯片中数据存储块的管理的方法, 尤其是一种闪存块控制区标识的分区及闪存数据块读写信息状态标识的方法。

现有的闪存中的存储块 (Block) 中均包含数据存储区和坏块标识的控制区部分, 如图 1 所示。当需要对闪存数据区进行读、写操作时, 通过对  
10 相应坏块标识部分中的信息进行读取和分析避免对坏块进行读、写操作。一般生产出来时, 即在不良数据块号码存储区表内标识出了坏块的号码, 以便在向数据块写入数据时, 首先在不良数据块号码存储区表内查找是否有对应号码, 如果有, 则不能写入数据, 如果没有表示该数据块为可以写入的。但是, 上述的标识方法, 仅仅用于标识不能使用的坏块, 对于可用  
15 的存储块的使用状态、写入过程以及在其内存储的数据的有效性等信息均无法反映。

此外, 现有的闪存的数据更新和数据直接写入之间的速度差别非常大。一般而言, 对闪存中存储的数据进行更新操作所花费的时间要远远大于向闪存中写入同样数量数据所花费的时间。当需要对闪存中大量的数据进行  
20 更新时, 往往会因擦除再写入操作时间过长, 而影响相应系统的其他操作。这一现象在对以闪存为主要存储介质的计算机存储装置进行数据更新操作时尤为明显。对闪存进行更新操作之前, 必须首先对该存储块作一次有效的擦除, 然后才能向该存储块写入相应的数据。

闪存更新操作中数据擦除、写入动作是非均匀性, 不能控制管理, 一  
25 至是随机的, 这样必然导致闪存器件中各个存储块使用频度的不均匀, 这带来了其老化速度的不均匀, 进而影响整个器件的寿命。





有鉴于此，本发明的主要目的在于提供一种闪存中存储块的分区及读写信息标识的方法，对闪存各个存储块的管理控制区进行合理的划分，经划分后，该控制区不仅可以随时标识、记录坏块，还可以标识相应存储块的使用状态、写入过程以及在其内存储的数据的有效性等信息，为合理、  
5 有效地利用闪存中的各存储块提供技术手段。

为实现上述目的，本发明提供一种闪存中存储块的分区及读写信息标识的方法，闪存内部的各存储块（Block）均划分有数据存储区和控制区。其中控制区划分为索引区、逻辑地址指针区和状态标识区。

所述的状态标识区标识所对应存储块工作状态的状态标识位组可为相  
10 对应的存储块（Block）的三种状态之一：“空”或“忙”或“释放”；“空”表示给数据块内没有数据，可以使用；“忙”表示该数据块已经被占用，不能再写入数据；“释放”表示该数据块可以被擦除。

所述的索引区中的索引位组的信息为与之相对应的存储块使用状态之一：“未用过”或“已使用”；“未用过”表示数据块内未写入数据，“已  
15 使用”表示数据块内写入数据后。

所述的逻辑地址指针区内的逻辑地址指针组信息为与之相对应的存储块的物理地址或表示多次更新后、按照更新次数顺序排列记录的逻辑地址。

所述的索引区中的索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量相等。

20 所述的状态标识区的状态标识位组的数量可与数据块的数量相等。

本发明通过将闪存内部的各存储块控制区划分为索引区、逻辑地址指针区和状态标识区，改变了现有的闪存存储块（Block）中控制区仅仅标识或记录已知的不能使用的坏块，而不能对可用的存储块部分的使用状态、写入过程以及在其内存储的数据的有效性等信息进行反映的情况的不足之  
25 处，从而为能够合理地利用闪存芯片提供了技术基础。

以下结合附图对本发明作进一步的详细说明：



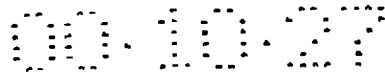


图 1 为现有闪存中存储块的数据存储区和管理区部分划分示意图。

图 2 为本发明对闪存中存储块的划分及状态信息标识的方法示意图。

图 3 为本发明对闪存中存储块首次写入数据后的状态示意图。

图 4 为本发明对闪存中存储块数据更新后的状态示意图。

5 如图 2 所示, 本发明为一种闪存中存储块的分区及读写信息标识的方法, 闪存内部的各存储块 (Block) 均划分有数据存储区和控制区, 数据区用来存放数据, 控制区用于标识对应的数据区的可用与否等状态信息。本发明的控制区划分为索引区、逻辑地址指针区和状态标识区。

10 状态标识区标识所对应存储块工作状态的状态标识位组可为相对应的存储块 (Block) 的三种状态之一: “空” 或 “忙” 或 “释放”。“空” 表示该数据块内用被占用, 可以写入数据, “忙” 表示该数据块被占用不能写入数据, “释放” 表示该数据块内的数据可以被擦除。当然, 这些状态标识可采用代码等区别符号描述。状态标识区的状态标识位组数与数据块的数量一致。

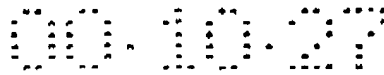
15 索引区中的索引位组的信息为与之相对应的存储块使用状态之一: “未用过” 或 “已使用”。其中 “未用过” 用 “1” 表示; “已使用” 用 “0” 表示。数据块内未写入数据, 索引位组的信息为 “0”; 数据块内写入数据后, 索引位组的信息为 “1”。这样标记为便于对应寻找数据更新后对应的数据块的逻辑地址指针。

20 逻辑地址指针区内的逻辑地址指针组信息为与之相对应的存储块的物理地址号码。索引区中的索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量相等。

1 号数据块为坏块, 该数据块的号码被记录在控制区不良存储表内。

25 当数据块内没有写入数据为空闲状态时, 索引位组的信息为 “1”, 状态标识位组可为 “空”, 表示该数据块为空闲状态, 逻辑地址指针组信息没有地址指针设置, 这是对应的状态标识位组为 “空”, 表示该数据块为空闲





状态，可写入数据。

参见图 3，在该实施例中，2 号数据块写入了数据，这是其对应的索引位信息置为 0，逻辑地址指针的地址置为该数据块的物理地址 2，2 号数据块的状态被标识为“忙”。

- 5 参见图 4，当对应的 2 号数据块有更新操作，并且更新的数据放入数据块状态标识位组的标记为“空”（如图 3 所示）的 3 号数据块中后，2 号数据块索引位组增加一个已使用的标记“0”，对应的增加标识逻辑地址指针区内的逻辑地址指针组信息也增加一个更新次数顺序记录的最后一个数据块逻辑地址，故应为“3”。这时，相应的 2 号数据块状态标识位组的标记为“释放”；3 号数据块状态标识位组的标记为“忙”。在某时刻，由系统对所用标识为“释放”的块进行擦除，擦除后，将该数据块的状态标识位组的标记为“空”，以备以后使用。因此数据更新时只要将数据直接写入一个空的数据块中即可，而不需要先进行擦除的操作，节省了大量的时间，提高了闪存的读写速度。另外，对于数据块的使用，也可由指针对空状态
- 10 的数据块——排序循环使用，保证数据块使用的均匀性。
- 15

因此，可知标识索引状态的索引区中的索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量相等。其索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量应为该数据块更新操作的次数。而状态标识区的状态标识位组的数量可与数据块的数量相等。

- 20 综上所述，本发明通过控制区的合理划分和标识记录，有效地实现了对闪存片数据块读写信息的管理，从而能够将对数据块写入和擦除动作分别完成，提高了数据读写的速度，而且能够均匀地使用闪存片个数据块，延长其使用寿命。



00-10-07

BEST AVAILABLE COPY

## 说明书附图

存储块编号	数据区	状态标识位组
#1	空	
#2	空	空
	空	空
	空	空
	空	空
#N	空	空

图 1

存储块编号	数据区	索引位组信息	逻辑地址指针组	状态标识位组
#1	空	1	空	
#2	空	1	空	空
	空	1	空	空
	空	1	空	空
	空	1	空	空
#N	空	1	空	空

图 2



00.10.27

BEST AVAILABLE COPY

存储块编号	数据区	索引位组信息	逻辑地址指针组	状态标识位组
#1	空	1	空	坏的
	空	1	空	空
#2	Data 2	0	2	忙
	空	1	空	空
	空	1	空	空
#N	空	1	空	空

图 3

存储块编号	数据区	索引位组信息	逻辑地址指针组	状态标识位组
#1	空	1	空	坏的
	空	1	空	空
#2	1st Data 2	0	2	释放
		0	3	
#3	2nd New Data 2	1	空	忙
	空	1	空	空
#N	空	1	空	空

图 4